Diabetes Data Analysis

FY 2016

Author Eric Lewis

TABLE OF CONTENTS

Contents

Introduction

The objective of this machine learning analysis is to examine the effects of ten baseline predictor variables in the diabetes dataset in Efron et al. (2003). The variables are: age, sex, body mass index (bmi), average blood pressure (map), and six blood serum measurements (tc, ldl, hdl, tch, ltg, glu). These variables are all quantitative in nature and measure the disease progression one year after the baseline. This analysis is outlined as follows an introduction, analysis, results, conclusion, and an appendix of R code.

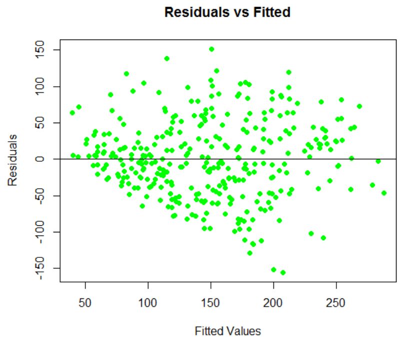
Analysis

This analysis begins with an initial examination of the diabetes dataset, which contains 442 patient observations, including their ten predictor variables. The dataset is randomly partitioned into two groups, training (75%), and testing (25%). The training data will be utilized to define the x and y training predictor matrix and the testing data will be utilized to define the x and y testing response matrix. The analysis phase of this project includes using Least Squares regression, best subset selection using BIC to select the number of predictors, best subset selection using 10-fold cross-validation to select the number of predictors, Ridge Regression modeling, and Lasso Regression modeling. Four of the models are compared using mean square error, (MSE), with the lowest value indicating the best fit model. The smaller the MSE, the closer the predicted responses are to the true responses.

Results

Least Squares Regression

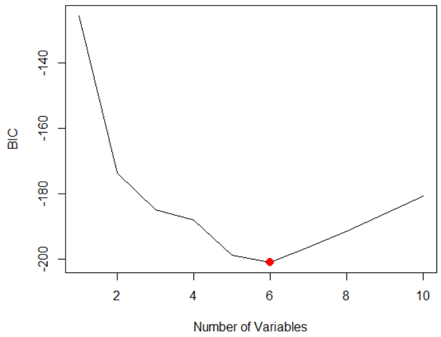
This results begin using least squares regression to model all ten predictors. This model returns a MSE of 3111.27. The expectation of this model is to be considered as a baseline model through the use of all ten predictors. The plot of residuals vs fitted values indicates the distribution of the residuals appears to be random, plot 1.



Plot 1: Residuals vs. Fitted

Best Subset Selection Using BIC

The next logical step is the determine which and how many of the ten predictors are the best to select for model development and comparison. The best subset selection using BIC to select the predictors is the following six predictors: Sex, BMI, MAP, TC, TCH, and LTG. The BIC for selecting the six variables is -201.13, Mallows C_p is 5.75, and the adjusted R^2 is 50.85%. The BIC, C_p are clear indictors that using six variables is the best model. The MSE for this 3095.48, which is 15.79 less than using the least squares regression model having ten variables. The decrease in the MSE indicates the quality of the estimator is better value as it approaches zero. An important consideration is that the adjusted R^2 is slightly better using seven variables; however, the difference is 0.0001 which is considered as insignificant as compared to clear results using BIC, and C_p. The selection is further demonstrated with the variable selection plot 2 and the table 1 of coefficients.



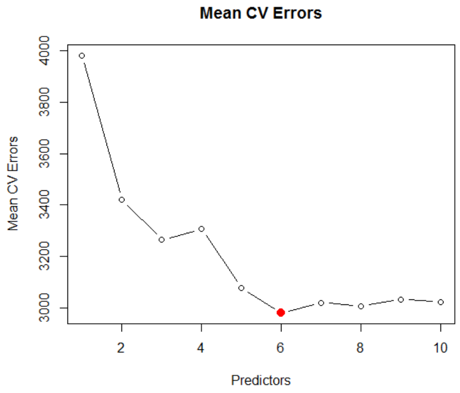
Plot 2: Variable Selection Using BIC

	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	150.12	2.96	50.68	2.3e-156
sex	-306.04	69.04	-4.43	1.3e-05
bmi	538.83	74.22	7.26	2.9e-12
map	389.07	69.92	5.57	5.5e-08
tc	-379.04	82.27	-4.61	5.9e-06
tch	332.67	88.97	3.74	2.2e-04
ltg	527.57	87.07	6.06	3.8e-09

Table 1: Variable Selection Coefficients

Best Subset Selection Using 10-Fold Cross-Validation

Six variables were chosen using BIC, the following best subset selection will validate that using six of the ten predictor variables is the best selection for model development. The 10-fold cross-validation will fold the data into 10 portions, using each fold as a validation until all of the MSE's are averaged. Plot 3 of the mean cross validation errors and table 2 of the coefficients agrees that the six variable selection model using BIC is the best predictors for modeling the diabetes data set. The MSE for the 10-fold cross-validation is 3095.48, the same as the best subset using BIC.



Plot 3: Mean Cross-Validation Errors

Intercept	Sex	BMI	MAP	TC	TCH	LTG
150.12	-306.04	538.83	389.07	-379.04	332.67	527.57

Table 2: Variable Selection Coefficients

Ridge Regression Using 10-Fold Cross-Validation

Ridge Regression is very similar to least squares regression with the exception that the coefficients are estimated by minimizing a different quantity. Ridge Regression does improve over least squares in the manner that least squares are based upon a bias-variance trade-off. As Lambda increases, the flexibility of the ridge regression fit decreases, leading to a decreased variance; however, an increased bias. The MSE for ridge regression is 3070.64 which is a 40.63 decrease over Least Squares Regression, thus demonstration model improvement. Table 3 of Ridge Regression coefficients demonstrates the model is using all ten variables as coefficients, which will be compared and contrasted with Lasso Regression.

Predictors	Coefficients
age	-11.28
sex	-156.90
bmi	374.45
map	264.88
tc	-32.00
ldl	-66.94
hdl	-173.94
tch	123.99
ltg	307.69
glu	134.50

Table 3: Ridge Coefficients

Lasso Model Using 10-Fold Cross-Validation

Lasso Model is will include all the predictors in the model as the Ridge Regression performed in this analysis. The Lasso Regression Model has an advantage over the Ridge Regression in the manner that it produces a simpler and more interpretable model. Lasso Regression does outperform the Ridge Regression using the Diabetes dataset based upon the Lasso Model is zeroing out age, tc, ldl, and tch, leaving six variables in the model, as shown in tables 4 and 5.

Predictors	Coefficients
age	0.00
sex	-119.62
bmi	501.57
map	270.93
tc	0.00
ldl	0.00
hdl	-180.29
tch	0.00
ltg	390.55
glu	16.59

Table 4: All Lasso Coefficients

Predictors	Coefficients
sex	-119.62
bmi	501.57
map	270.93
hdl	-180.29
ltg	390.55
glu	16.59

Table 5: Lasso Coefficients

Conclusion

The results of the diabetes dataset analysis demonstrate the best model is using the Lasso Regression having a MSE of 2920.04 which is 150.06 lower than the Ridge Regression Model. The Lasso Regression Model also utilized six predictors as the best subset selection using BIC and 10-fold cross-validation confirmed. It is expected that the Lasso Regression Model outperforms the Ridge Regression Model in a setting where there are ten predictors have substantial coefficients that are very small of equal to zero. Ridge Regression would have performed better when the response is a function of many predictors having coefficients of equal size. This analysis does conclude that the that diabetes progression is predictable using the diabetes dataset and is best suited for modeling using the Lasso Regression Model.

Model	MSE	$Adjr^2$	C_p	BIC
Least Squares Regression	3111.27			
Best subset selection using BIC {6}	3095.48	50.85%	5.75	-201.13
Best subset selection using 10-fold Cross-Validation {6}	3095.48			
Ridge Regression Using 10-Fold Cross-Validation	3070.64			
Lasso Regression Using 10-Fold Cross-Validation	2920.04			

Table 6: Summary of Key Statistics

Appendix

R Code

```
# (0) Load the diabetes data

library(lars)
data(diabetes)
data.all <- data.frame(cbind(diabetes$x, y = diabetes$y))
```

```
# (0) Show contents, summary of data, and check for NA in the data.

str(data.all)
summary(data.all)
sum(is.na(data.all))
```

```
# (0) Partition the patients into two groups: training (75%) and test (25%)

n <- dim(data.all)[1] # sample size = 442
set.seed(1306) # set random number generator seed to enable

# (0) repeatability of results
test <- sample(n, round(n/4)) # randomly sample 25% test
data.train <- data.all[-test,]
data.test <- data.all[test,]
x <- model.matrix(y ~ ., data = data.all)[,-1] # define predictor matrix

# (0) excl intercept col of 1s
x.train <- x[-test,] # define training predictor matrix
x.test <- x[test,] # define test predictor matrix
y <- data.all$y # define response variable
y.train <- y[-test] # define training response variable
y.test <- y[test] # define test response variable
n.train <- dim(data.train)[1] # training sample size = 332
n.test <- dim(data.test)[1] # test sample size = 110
```

```
# (1) Least squares regression using all 10 predictors (Function LM)

lm.full = lm(y~. , data = data.train)
pred = predict(lm.full, data.test)


plot(lm.full$fitted.values,resid(lm.full), xlab="Fitted Values", ylab="Residuals",
main="Residuals vs Fitted", pch=19, col="green")
abline(h=0)


msefull = mean((y.test - pred)^2)
print(paste("MSE for full model: ",msefull))
```

# (2) Best subset selection using BIC to select predictors (Package Leaps, regsubsets)

```
library(leaps)


regfit.full = regsubsets(y~. , data = data.train , nvmax = 10)
reg.summary = summary(regfit.full)
reg.summary


reg.summary$bic
which.min(reg.summary$bic)

reg.summary$cp
which.min(reg.summary$cp)

reg.summary$adjr2
which.max(reg.summary$adjr2)
reg.summary$adjr2[7]-reg.summary$adjr2[6]


plot(reg.summary$adjr2, xlab="Number of Variables",ylab="BIC",type='l')
points(6,reg.summary$adjr2[6],col="red",cex=2,pch=20)
points(6,reg.summary$adjr2[7],col="blue",cex=2,pch=20)

plot(reg.summary$bic, xlab="Number of Variables",ylab="BIC",type='l')
points(6,reg.summary$bic[6],col="red",cex=2,pch=20)

print(paste("Best subset is 6 having a BIC: ", reg.summary$bic[6]))

cat(names(coef(regfit.full,6)))
coef(summary(lm(y~sex+bmi+map+tc+tch+ltg, data = data.train)))

coefi  = coef(regfit.full,id=6)
pred   = coefi[1] + (x.test[, names(coefi[-1])]%*%coefi[-1])
msesub = mean((y.test - pred)^2)
print(paste("MSE Best Subsets (BIC) ",msesub))
```

# (3) Best subset selection using 10-fold cross-validation

```
predict.regsubsets = function(object,newdata,id,...){
        form=as.formula(object$call[[2]])
        mat = model.matrix(form, newdata)
        coefi = coef(object, id=id)
        xvars = names(coefi)
        mat[,xvars]%*%coefi
}

k=10
set.seed(1306)

folds = sample(1:k, nrow(data.train),replace = TRUE)
cv.errors = matrix(NA,k,10, dimnames = list(NULL, paste(1:10)))

for (j in 1:k){
        best.fit = regsubsets(y~. ,data=data.train[folds!=j,],nvmax = 10)
        for (i in 1:10){
                pred = predict.regsubsets(best.fit, data.train[folds==j,], id=i)
                cv.errors[j,i] = mean((data.train$y[folds==j]-pred)^2)
        }
}

mean.cv.errors = apply(cv.errors, 2, mean)
mean.cv.errors


plot(mean.cv.errors, type="b", main = "Mean CV Errors",xlab = "Predictors",
     ylab="Mean CV Errors")
which.min(mean.cv.errors)
points(6,mean.cv.errors[6],col="red",cex=2,pch=20)


regfit.cv = regsubsets(y~. , data = data.train , nvmax = 10)
coefi  = coef(regfit.cv,id=6)
pred   = coefi[1] + (x.test[, names(coefi[-1])]%*%coefi[-1])
coefi

msebestsub = mean((y.test - pred)^2)
print(paste("MSE Best subsets ",msebestsub))
```

4) Ridge regression using 10-fold cross-validation

```
library(glmnet)

set.seed(1306)
grid = 10^seq(10,-2,length=100)

cv.out = cv.glmnet(x.train, y.train,alpha=0)
bestlam = cv.out$lambda.1se
bestlam

ridge.mod = glmnet(x.train,y.train,alpha=0,lambda=grid,thresh=1e-12)
ridge.pred = predict(ridge.mod,s=bestlam,newx=x.test)


mseridge = mean((y.test - ridge.pred)^2)
print(paste("MSE Ridge ",mseridge))

coeff = glmnet(x.train,y.train,alpha=0,lambda=bestlam,thresh=1e-12)$beta
matrix(coeff,dimnames=list(row.names(coeff),c("Coefficients")))
```

# (5) Lasso Model using 10-fold cross-validation

```
set.seed(1306)

cv.out = cv.glmnet(x.train,y.train,alpha=1)
bestlam = cv.out$lambda.1se
bestlam

lasso.mod = glmnet(x.train,y.train,alpha=1,lambda=bestlam)
lasso.pred = predict(lasso.mod,s=bestlam,newx=x.test)


mselasso = mean((y.test-lasso.pred)^2)
print(paste("MSE Lasso ",mselasso))


##Summary(coeffsummary = glmnet(x.train,y.train,alpha=1,lambda=bestlam)$beta

coeff = glmnet(x.train,y.train,alpha=1,lambda=bestlam)$beta
matrix(coeff,dimnames=list(row.names(coeff),c("Coefficients")))
#lasso.coef[lasso.coef!=0]


lasso.mod = glmnet(x.train,y.train,alpha=1,lambda=grid)
plot(lasso.mod)
plot(cv.out)
```

# (5) Lasso Model using Covariance Test

```
# Computes covariance test for adaptive linear modelling
# Gaussian

library(covTest)

a=lars(x.train,y.train)
covTest(a,x.train,y.train)
```